am

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/693,090	10/20/2000	Jennifer Q. Trelewicz	BLD9-2000-0060-US2	7807

7590 06/01/2005

John L Rogitz
Rogitz & Associates
750 B Street Suite 3120
San Diego, CA 92101

EXAMINER
KISS, ERIC B

ART UNIT	PAPER NUMBER
2192	

DATE MAILED: 06/01/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

MAILED
JUN 0 1 2005
Technology Center 2100

Application Number: 09/693,090
Filing Date: October 20, 2000
Appellant(s): TRELEWICZ ET AL.

John L. Rogitz
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 7 December 2004.

(1) REAL PARTY IN INTEREST

A statement identifying the real party in interest is contained in the brief.

(2) RELATED APPEALS AND INTERFERENCES

A statement identifying the related appeals and interferences which will directly affect or be directly affected by or have a bearing on the decision in the pending appeal is contained in the brief.

(3) STATUS OF CLAIMS

The statement of the status of the claims contained in the brief is correct.

(4) STATUS OF AMENDMENTS

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) SUMMARY OF CLAIMED SUBJECT MATTER

The summary of the claimed subject matter contained in the brief is correct.

(6) GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

The appellant's statement of the grounds of rejection in the brief is correct.

The following grounds of rejection are applicable to the appealed claims:

Claims 6, 9, 11-18, 21, 23, 25, 26, and 29 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. This rejection is set forth in a prior Office Action, mailed on 1 September 2004, and has be reproduced it item (10) below.

Claims 1-6, 11-21, and 23-26 are rejected under 35 U.S.C. 102(b) as being anticipated by [Fi9D98]. This rejection is set forth in a prior Office Action, mailed on 1 September 2004, and has be reproduced it item (10) below.

Claims 5 and 29 are rejected under 35 U.S.C. 103(a) as being unpatentable over [FiD98] in view of *Staugaard, Jr.* This rejection is set forth in a prior Office Action, mailed on 1 September 2004, and has be reproduced it item (10) below.

Claims 7-10, 22, 27, and 28 are rejected under 35 U.S.C. 103(a) as being unpatentable over [FiD98] in view of U.S. Patent No. 6,080,204 to Mendel. This rejection is set forth in a prior Office Action, mailed on 1 September 2004, and has be reproduced it item (10) below.

(7) CLAIMS APPEALED

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) PRIOR ART OF RECORD

a) Randall J. Fisher and Henry G. Dietz, "Compiling for SIMD Within a Register," 1998 Workshop on Languages and Compilers for Parallel Computing, North Carolina, Aug 1998 (hereinafter [FiD98]).

b) Andrew C. Staugaard, Jr., "Structured and Object-Oriented Techniques: An Introduction Using C++," 1997, Prentice-Hall, Inc. (hereinafter *Staugaard, Jr.*).

c)	6,080,204	MENDEL	6-2000

(9) RESPONSE TO ARGUMENT

(9a) The rejection of claims 6, 9, 11-18, 21, 23, 25, 26, and 29 under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement

Appellant's specification provides, on page 4, in lines 6-9, a lexicon containing definitions of the terms ***input precision*** and ***output precision***. In particular, the originally filed specification stated the following:

> In this disclosure, "input precision" is used to reference the initial precision of individual data elements prior to simultaneous operation, while "output precision" is used to reference the final maximum precision of individual data elements prior to simultaneous operations have [sic] been completed.

In the response filed 2 June 2004, Appellant significantly altered the definition of the term ***input precision*** by replacing the phrase "prior to" with its opposite, "after". No evidence has been presented that this amendment has been made merely to correct any obvious error. The section of the specification that Appellant refers to (page 4, lines 15-20), in an attempt to justify the amendment, merely describes how output precision can be determined based on the known input precision. In other words, the manipulations described are in no way reflected in either the original or the modified definition of ***input precision***. Accordingly, claims 6, 9, 11-18, 21, 23, 25, 26, and 29, which recite (or are dependent from a parent or base claim that recites) the term ***input precision***, contain new matter not embraced by the originally filed disclosure.

(9b) The rejection of claims 1-6, 11-21, and 23-26 under 35 U.S.C. 102(b) as being anticipated by [FiD98]

Claim 1 recites, "...simultaneously processing the elements, wherein at least one of: a carry, and a borrow, **may** occur between data elements in a register" [emphasis added]. The occurrence of a carry or borrow is recited using optional language. As such, the claim appears to cover both a system where such a carry or borrow **do** occur and a system where such a carry or borrow **do not** occur. Therefore, [FiD98], in which the carry or borrow do not occur, still anticipates claim 1.

Further, even if the optional language of claim 1 is interpreted as required language, the affected clause should not be given significant patentable weight as it does not properly modify the corresponding limitation of "simultaneously processing the elements" by limiting or describing how the simultaneous processing is carried out. As Appellant is apparently not seeking patent protection on an invention that requires carrying or borrowing between data elements in a register, a prior art reference should likewise not be required to exhibit such behavior in order to anticipate claim 1.

(9c) The rejection of claims 5 and 29 under 35 U.S.C. 103(a) as being unpatentable over [FiD98] in view of *Staugaard, Jr.*

The Examiner maintains that *Staugaard, Jr.* provides an extensive teaching of why program documentation in the form of comments is important in software development. Specifically, page 75 of *Staugaard, Jr.* is replete with motivating reasons to add such comments to several portions of source code. Among the reasons given is the following:

> Remember, someone (including you) might have to debug or maintain the program in the future. A good commenting scheme makes these tasks a much more efficient and pleasant process.

The final sentence of section 7c of Appellant's brief is unrelated to the ground of rejection at issue and is ignored.

(9d) The rejection of claims 7-10, 22, 27, and 28 under 35 U.S.C. 103(a) as being unpatentable over [FiD98] in view of U.S. Patent No. 6,080,204 to Mendel.

As Appellant has failed to provide any specific arguments traversing this ground of rejection , this rejection should be sustained.

(10) The rejections of claims 1-29, as stated in the Final Rejection mailed 1 September 2004, on pages 7-16, are reproduced below for completeness.

Claims 6, 9, 11-18, 21, 23, 25, 26, and 29 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention. As discussed above, in the amendment to the first paragraph of p. 4, Applicant significantly alters the definition of the term "input precision" by replacing the phrase "prior to" with its opposite, "after". No evidence has been presented that this amendment has been made merely to correct any obvious error. Accordingly, claims 6, 9, 11-18, 21, 23, 25, 26, and 29, which recite (or are dependent from a parent or base claim that recites) the term "input precision", contain new matter not embraced by the originally filed disclosure.

In the interest of compact prosecution, the Examiner maintains the previous interpretation of the term "input precision" as indicated by the rejections below.

Claims 1-6, 11-21, and 23-26 are rejected under 35 U.S.C. 102(b) as being anticipated by Randall J. Fisher and Henry G. Dietz, "Compiling for SIMD Within a Register," 1998 Workshop on Languages and Compilers for Parallel Computing, North Carolina, Aug 1998 (hereinafter [FiD98]).

The detailed discussion of the individual rejected claims below is presented out of sequence to better reflect the dependency groupings of the claims. The order in which the claims are addressed represents the general order in which the claims would be renumbered by the Examiner upon allowance of the application.

As per claim 1, [FiD98] discloses a compiler receiving higher-level code and outputting lower-level code to enable a processor to simultaneously process multiple multi-bit data elements in a single register (see, for example, section 1 on pp. 1-4), the logic of the lower-level code including: establishing at least first and second signed, multi-bit data elements in at least a first register (see, for example, see, for example, section 1 on pp. 1-4; Note that the SWARC language supports signed data types--see, for example, section 4.1.1 on p. 15); and simultaneously processing the elements (see, for example, section 1 on pp. 1-4). Applicant's newly added claim limitations are specified in permissive language (for example, reciting "may"), and the broadest reasonable interpretation of these limitations is that they are optional components.

As per claim 2, [FiD98] further discloses the compiler accessing at least one of: a compiler directive, a flag, or a configuration file, to decide when to make elements independent of each other (see, for example, section 4 on pp. 14-18).

As per claim 19, [FiD98] further discloses the compiler generating instructions to pack multiple data elements from respective data sources into a common register to be operated on by an algorithm simultaneously with each other (see, for example, section 4 on pp. 14-18).

As per claim 20, [FiD98] further discloses the first element being a first partial element having a related second partial element established in a second register, and the lower-level code output by the compiler causing the first and second partial elements to be combined after processing (see, for example, section 2.3 on p. 9).

As per claim 3, [FiD98] further discloses a first element being provided from a first data set and a second element being provided from a second data set different from the first (see, for example, section 1 on pp. 1-4; and section 2 on pp. 4-11).

As per claim 4, [FiD98] further discloses the compiler allocating a respective output precision in a register for each data element to be processed in the register during a single cycle (see, for example, section 4.1.1 on p. 15).

As per claim 23, [FiD98] further discloses the compiler determining the output precision based at least in part on an input precision (see, for example, section 4.1.1 on p. 15).

As per claim 25, [FiD98] further discloses the compiler adding a bit of precision if the maximum magnitude negative number that is required for the data during processing is the maximum negative number that can be represented in the respective precision (see, for example, section 2.1.2 on pp. 6-7; and section 3.1 on p. 11).

As per claim 26, [FiD98] further discloses the compiler adding at least one bit of precision based at least in part on an operation on a data element (see, for example, section 4.1.1 on p. 15).

As per claim 24, [FiD98] further discloses the compiler receiving, as input, the output precision (see, for example, section 4.1.1 on p. 15).

As per claim 6, [FiD98] further discloses an output precision or an input precision being defined by means of a compiler directive, or a configuration file, or a variable definition (see, for example, section 4.1.1 on p. 15).

As per claim 11, [FiD98] discloses defining at least one compiler directive, instructions, or configuration file for a compiler defining at least one of: an input precision for at least one data element (see, for example, section 4.1.1 on p. 15); and multiple data sources of respective data elements to be packed into a common register and operated on by an algorithm simultaneously with each other (see, for example, section 1 on pp. 1-4; and section 2 on pp. 4-11). Applicant's newly added claim limitations are specified in permissive language (for example, reciting "can"), and the broadest reasonable interpretation of these limitations is that they are optional components.

As per claim 12, [FiD98] further discloses the compiler determining first and second precisions to be allocated in a single register to hold respective first and second signed data elements (see, for example, section 1 on pp. 1-4; and section 2 on pp. 4-11), and the compiler generating a lower-level code from a higher level code to undertake method acts comprising: packing the elements into the register (see, for example, section 4 on pp. 14-18); and operating on the elements (see, for example, section 1 on pp. 1-4; and section 2 on pp. 4-11).

As per claim 13, [FiD98] further discloses the register sending plural data elements simultaneously to at least one computational subsystem (see, for example, section 1 on pp. 1-4; and section 2 on pp. 4-11).

As per claim 14, [FiD98] further discloses the operation being a multiplication by a constant or by a variable of known precision, or an addition, or a shift-left logical, or a subtraction, or a bitwise AND, or a bitwise OR (see, for example, section 1 on pp. 1-4; and section 2 on pp. 4-11).

As per claim 15, [FiD98] further discloses the elements being independent of each other as defined by the compiler directive or configuration file (see, for example, section 1 on pp. 1-4; section 2 on pp. 4-11; and section 4 on pp. 14-18).

As per claim 16, [FiD98] further discloses the first element being provided from a first data set and the second element being provided from a second data set different than the first (see, for example, section 1 on pp. 1-4; and section 2 on pp. 4-11).

As per claim 17, [FiD98] further discloses the first element being a first partial element having a related second partial element established in a second register, and the lower-level code causing the first and second partial elements to be combined after processing (see, for example, section 1 on pp. 1-4; and section 2 on pp. 4-11).

As per claim 18, [FiD98] further discloses determining first and second precisions including determining the precisions such that the maximum negative number that can be represented in an element is one larger that the maximum negative number that can be represented in the respective precision (see, for example, section 2.1.2 on pp. 6-7; and section 3.1 on p. 11).

As per claim 21, [FiD98] further discloses the compiler directive, instructions, or configuration file embodying instructions to compile predetermined portions of code received by the compiler to be executed simultaneously on packed data (see, for example, section 1 on pp. 1-4; section 2 on pp. 4-11; and section 4 on pp. 14-18).

Claims 5 and 29 are rejected under 35 U.S.C. 103(a) as being unpatentable over [FiD98], as applied to claims 1 and 11 above in view of Andrew C. Staugaard, Jr., "Structured and Object-Oriented Techniques: an Introduction Using C++," 1997, Prentice-Hall, Inc. (hereinafter *Staugaard, Jr.*).

As per claims 5 and 29, [FiD98] in addition to the disclosure applied above, [FiD98] fails to expressly disclose the compiler receiving instructions not to compile a predetermined portion of code received by the compiler. However, *Staugaard, Jr.* teaches that comment statements have been well known and used in the art of computer programming to provide insightful documentation making source code easier to read by humans. *Staugaard, Jr.* further teaches that such comment statements are not compiled, *e.g.*, everything following double forward slashes ("//") in a line of C++ source code is ignored by a C++ compiler (see, for example, p.70, paragraph 3; and p. 75, whole page). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the system of [FiD98] to include instructions not to compile a predetermined portion of code. One would be motivated to do so to provide additional documentation without changing the behavior of a program.

Claims 7-10, 22, 27, and 28 are rejected under 35 U.S.C. 103(a) as being unpatentable over [FiD98], as applied to claims 1 and 11 above in view of U.S. Patent No. 6,080,204 to Mendel.

As per claim 7, [FiD98] discloses a compiler program for outputting lower-level code to process multi-bit, signed data elements (see, for example, see, for example, section 1 on pp. 1-4; Note that the SWARC language supports signed data types--see, for example, section 4.1.1 on p. 15), the lower-level code comprising: packing at least first and second data elements into a single register (see, for example, section 1 on pp. 1-4); and processing the elements simultaneously (see, for example, section 1 on pp. 1-4). [FiD98] fails to expressly disclose the use of a computer program storage device readable by a digital processing apparatus for implementing the prescribed functions. However, Mendel teaches that it has been known to use such computer program storage devices for storing computer program instructions to implement a compiler (see, for example, col. 30, line 60, through col. 34, line 16). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the system of [FiD98] to include the use of such a device in order to provide a means of storing/transporting/executing instructions necessary for implementing the disclosed computer-enabled procedure. Applicant's newly added claim limitations are specified in permissive language (for example, reciting "permitted to"), and the broadest reasonable interpretation of these limitations is that they are optional components.

As per claim 8, [FiD98] further discloses flag means indicating whether a precision should be checked in at least one cycle (see, for example, section 4.1.1 on p. 15). Therefore, for reasons stated above, such a claim also would have been obvious.

As per claim 9, [FiD98] further discloses compiler directive means for defining an input precision (see, for example, section 4.1.1 on p. 15). Therefore, for reasons stated above, such a claim also would have been obvious.

As per claim 10, [FiD98] further discloses compiler directive means for defining multiple data sources of respective data elements to be packed into a common register and operated on by an algorithm simultaneously with each other (see, for example, section 1 on pp. 1-4; and section 2 on pp. 4-11). Therefore, for reasons stated above, such a claim also would have been obvious.

As per claim 22, [FiD98] further discloses means for indicating whether a precision should be checked (see, for example, section 4.1.1 on p. 15); means responsive to the means for indicating for checking that the packed elements do not overflow or underflow or achieve a maximum magnitude negative number representable in the precision in a cycle, undertaking wrap or saturation in the elements to prevent corruption of other data elements in a register, or signaling an error to be handled by an error-handling routine in the program (see, for example, section 1 on pp. 1-4). Therefore, for reasons stated above, such a claim also would have been obvious.

As per claim 27, [FiD98] further discloses means for adding a bit of precision if the maximum magnitude negative number that is required for the data during processing is the maximum magnitude negative number that can be represented in the respective precision (see, for example, section 2.1.2 on pp. 6-7; and section 3.1 on p. 11). Therefore, for reasons stated above, such a claim also would have been obvious.

As per claim 28, [FiD98] further discloses means for adding at least one bit of precision based at least partially on an operation on a data element (see, for example, section 4.1.1 on p. 15). Therefore, for reasons stated above, such a claim also would have been obvious.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

EBK
May 31, 2005

TUAN DAM
SUPERVISORY PATENT EXAMINER

Conferees
Tuan Q. Dam
Kakali Chaki

Kakali Chaki

KAKALI CHAKI
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100

John L Rogitz
Rogitz & Associates
750 B Street Suite 3120
San Diego, CA 92101